Optimal transportation problems with connectivity constraints

Frédéric Cazals, Dorian Mazauric

Optimal transportation problems with connectivity constraints

Frédéric Cazals and Dorian Mazauric

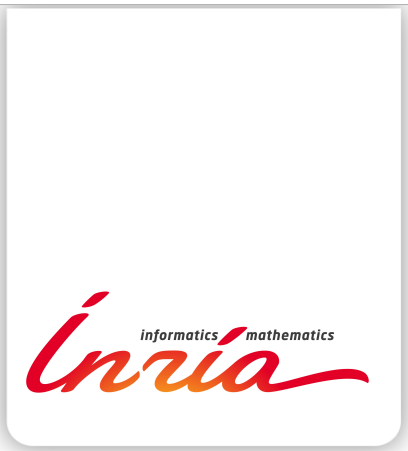
RESEARCH REPORT

N° 8991

Décembre 2016

Project-Team Algorithms-Biology-Structure

ISSN 0249-6399 ISRN INRIA/RR--8991--FR+ENG



Optimal transportation problems with connectivity constraints

Frédéric Cazals and Dorian Mazauric

Project-Team Algorithms-Biology-Structure

Research Report n° 8991 — Décembre 2016 — 24 pages

Abstract: The earth mover distance (`EMD`) or the Mallows distance are example optimal transportation (OT) problems reducing to linear programs. In this work, we study a generalization of these problems when the supply and demand nodes are the vertices of two graphs called the *supply* and the *demand* graphs. The novel problems embed *connectivity constraints* in the transport plans computed, using a Lipschitz-like condition involving distances between certain subgraphs of the supply graph and certain subgraphs of the demand graph. More precisely, we make three contributions.
First, we formally introduce two optimal transportation problems generalizing EMD, namely *Minimum-cost under flow, transport size, and connectivity constraints problem* (problem `EMD-FCC`) and *Maximum-flow under cost, transport size, and connectivity constraints problem* (problem `EMD-CCC`). We prove that problems `EMD-CCC` and `EMD-FCC` are NP-complete, and that `EMD-FCC` is hard to approximate within any given constant. Second, we develop a greedy heuristic algorithm returning admissible solutions, of time complexity $O(n^3m^2)$ with n and m the numbers of vertices of the supply and demand graphs, respectively. Third, on the experimental side, we apply our novel OT algorithms for two applications, namely the comparison of clusterings, and the analysis of so-called potential energy landscapes in molecular science. These experiments show that optimizing the transport plan and respecting connectivity constraint can be competing objectives. Implementations of our algorithms are available in the Structural Bioinformatics Library at `http://sbl.inria.fr`.

Key-words: Optimal transportation, connectivity constraints, NP-completeness, graph algorithms, clustering analysis, molecular simulation

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Transport optimal avec contraintes de connectivité

Résumé : La distance du transport de masse ou la distance de Mallows dérivent de problèmes de transport optimal, et leur calcul se réduit à un programme linéaire. Dans ce travail, nous étudions une généralisation de tels problèmes, dans le cas où les points de ressource et de demande sont les sommets de graphes dits de *ressource* et de *demande*. Les problèmes étudiés incorporent des contraintes de type Lipschitz entre certains sous-graphes de ces deux graphes. Plus précisément, nous effectuons trois contributions.

D'une part, nous introduisons deux problèmes de transport optimal, nommés *Minimum-cost under flow, transport size, and connectivity constraints problem* (probleme `EMD-FCC`) et *Maximum-flow under cost, transport size, and connectivity constraints problem* (probleme `EMD-CCC`). Nous prouvons que `EMD-CCC` and `EMD-FCC` sont NP-complets, et que `EMD-FCC` est difficile à approximer. D'autre part, nous développons une heuristique retournant des solutions admissibles en temps $O(n^3m^2)$ avec n et m le nombre de sommets des graphes de ressource et de demande. Enfin, d'un point de vue expérimental, nous utilisons nos algorithmes pour comparer des clusterings et pour analyser des paysages énergétiques moléculaires. Il apparaît que l'optimisation d'un plan de transport et le respect des contraintes de connectivité peuvent être des objectifs antagonistes. Nos implementations sont disponibles dans la *Structural Bioinformatics Library*, voir `http://sbl.inria.fr`.

Mots-clés : Transport optimal, contraintes de connectivité, NP-complétude, graphes et algorithmes, clustering, simulation moléculaire

Contents

1 Introduction

Optimal transportation (OT) problems have a long standing history in mathematics and computer science, originating with the works of Monge on earth moving (≪ *la théorie des déblais et des remblais* ≫) [12]. Such problems were later rephrased in terms of Riemannian geometry and measure theory [17], one key concept being the distance between two distributions, namely is the minimal amount of work that must be performed to transform one distribution into the other by moving distribution mass around. Various applications were developed across all sciences, one of the early ones in computer science being the *earth mover distance* (`EMD`), used to compare two images using their color histograms [15]. The EMD is also related to the Mallows distance used in statistics [11]. Both distances are of special interest since they can be phrased as linear programs, and hence, are amenable to efficient algorithms. Beyond linear programming, OT problems are generally hard to solve, which motivated the development of stochastic optimization techniques [8], or the exploitation of specific (geometry) properties of the functional studied [16].

In this work, we explore a new dimension of OT problems, namely when the supply and demand nodes are the vertices of two graphs called the *supply* and the *demand* graphs. In the presence of connectivity information provided by these graphs, it is natural to expect transport plans to *comply* with this connectivity information, while still minimizing the transport cost. For example, a transport plan may be termed valid provided that any connected component of the supply graph induces (via edges carrying flow) a connected component of the demand graph–a condition called *strict connectivity constraints* in the sequel. Naturally, depending on the nature of the constraints, one may end up with more costly transport plans, or worse, may face unfeasible problems. This paper precisely addresses these problems, the notion of connectivity constraints being modeled by a Lipschitz-like condition involving distances between certain subgraphs of the supply graph and certain subgraphs of the demand graph.

Practically, we apply our novel OT algorithms for two applications, namely the comparison of clusters obtained by mode seeking algorithms [5, 4], and also the analysis of potential energy landscapes (PEL) of molecular systems [18, 3, 1].

Paper overview and contributions. Section 2 introduces OT problems with connectivity constraints. In Section 3, we prove that the problems are NP-complete and not in APX for very simple classes of instances, and we prove stronger NP-completeness results for strict connectivity constraints. In Section 4, we develop exact, approximation and heuristic polynomial time algorithms, and we investigate the case of strict connectivity constraints. Finally, Section 5 present experiments on the comparison of clusterings and on the analysis of molecular potential energy landscapes.

Due to the lack of space, all proofs are presented in appendix.

2 Problem formulation and models

We first define notations, and proceed with various constraints (connectivity constraints, transport size), from which our optimal transportation problems are defined.

2.1 Pre-requisites

Notations. Consider two graphs: a *supply* graph $G = (V, E)$ and a *demand* graph $G' = (V', E')$. The set $V = \{v_1, \ldots, v_{|\mathcal{I}|}\}$ represents the supply nodes. We denote by $\mathcal{I}$ the set of indices of the supply nodes. The value $X_{v_i} \geq 0$ represents the volume of supply of node v_i for

all $i \in \mathcal{I}$. The set $V' = \{v'_1, \ldots, v'_{|\mathcal{J}|}\}$ represents the demand nodes. We denote by $\mathcal{J}$ the set of indices of the demand nodes. The value $Y_{v'_j} \geq 0$ represents the volume of demand of node v'_j for all $j \in \mathcal{J}$. Let $B = (V \cup V', V \times V')$ be the complete bipartite graph between the supply and the demand nodes. The real values $c_{v_i,v'_j} \geq 0$ represent the linear cost of sending a unit of flow from node v_i to node v'_j for all $i \in \mathcal{I}, j \in \mathcal{J}$. The variable f_{v_i,v'_j} represents the volume of flow sent by node v_i to v'_j for all $i \in \mathcal{I}$, $j \in \mathcal{J}$. When there is no ambiguity, we abuse the notation writing X_i, Y_j, $c_{i,j}$, and $f_{i,j}$ instead of X_{v_i}, $Y_{v'_j}$, c_{v_i,v'_j}, and f_{v_i,v'_j}, respectively. For all $i \in \mathcal{I}$, $j \in \mathcal{J}$, the cost of sending a volume of flow $f_{i,j}$ through edge $\{v_i, v'_j\} \in E(B)$ is $f_{i,j}c_{i,j}$. Given the flows $f_{i,j}$ for all edges of B, the total flow is $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}$ and the total cost is $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}c_{i,j}$.

Minimum-cost flow problem. The classical *minimum-cost flow problem* (`EMD`), or *transportation problem* [6], consists in determining a minimum-cost flow satisfying the demands and respecting the supply constraints. This problem is polynomial since it reduces to solving the following linear program (LP):

$$\begin{cases} \text{Minimize} \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}\ c_{i,j} \\ \text{subject to} \\ \sum_{i\in\mathcal{I}} f_{i,j} \leq Y_j \quad \forall j \in \mathcal{J}, \\ \sum_{j\in\mathcal{J}} f_{i,j} \leq X_i \quad \forall i \in \mathcal{I}, \\ \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} = \min(\sum_{i\in\mathcal{I}} X_i, \sum_{j\in\mathcal{J}} Y_j), \\ f_{i,j} \geq 0 \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}. \end{cases} \tag{1}$$

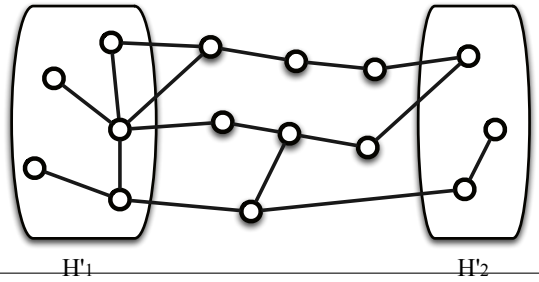
The first line is the objective function, the third line represents the demand constraints, the fourth line describes the supply constraints, the fifth line states that the total amount of flow equals the minimum between the total volume of supplies and the total volume of demands, and the last line guarantees that flows are positive. Note that if $\sum_{i\in\mathcal{I}} X_i \geq \sum_{j\in\mathcal{J}} Y_j$, then the fifth line can be removed and the inequality constraints of the third line become equality constraints. The number of edges that support flow is at most the total number of nodes (of G and G') minus one (e.g. see [9]). Say otherwise, $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}|f_{i,j}>0} 1 \leq |\mathcal{I}| + |\mathcal{J}| - 1$. Given an optimal solution f for `EMD`, we define the *total number of edges* M_{EMD}, the *total flow* F_{EMD}, the *total cost* C_{EMD}, and the *ratio* d_{EMD} (a.k.a. the *earth mover distance* [15]):

$$M_{\text{EMD}} = \sum_{i\in\mathcal{I},j\in\mathcal{J}} \mathbf{1}_{f_{i,j}>0}, F_{\text{EMD}} = \sum_{i\in\mathcal{I},j\in\mathcal{J}} f_{i,j}, C_{\text{EMD}} = \sum_{i\in\mathcal{I},j\in\mathcal{J}} f_{i,j}\ c_{i,j}, d_{\text{EMD}} = C_{\text{EMD}}/F_{\text{EMD}}. \tag{2}$$

Connectivity constraints. When the supply nodes (and likewise the demand nodes) are endowed with a graph structure, the distances between different sub-graphs of G must be correlated with the distances between the different sub-graphs of G' that receive flow from the former sub-graphs.

To formalize this idea, we introduce the following notations (Fig. 1). Let $d_{G'}(v, v')$ be the number of edges minus one of a minimum shortest path between $v \in V'$ and $v' \in V'$ in G'. If $v = v'$, we set $d_{G'}(v, v') = 0$. In the following, we denote by $cc(G')$ the set of maximal connected components of G'. Let $H' = \{H'_1, \ldots, H'_t\}$ be any set of $t \geq 1$ disjoint sub-graphs of G'. Note that H'_i is not necessarily a connected sub-graph. We define $d_{G'}(H'_i, H'_j) = \min_{v\in V(H'_i), v'\in V(H'_j)} d_{G'}(v, v')$. Note that if v and v' are not connected in G, then $d_{G'}(v, v') = \infty$. We define $d_{G'}(H') = \max_{i,j\in\{1,\ldots,t\}} d_{G'}(H'_i, H'_j)$. If $|H'| = 1$, then $d_{G'}(H') = 0$. Furthermore, we define $d_{G'}(cc(H'_i))$ as follows. Let $cc(H'_i) = \{cc_1(H'_i), \ldots, cc_s(H'_i)\}$ be the set of the $s \geq 1$ maximal connected

Figure 1 Distances between subgraphs. Example of graph G' with $H' = \{H'_1, H'_2\}$: $d_{G'}(H') = d_{G'}(H'_1, H'_2) = 1$, $d_{G'}(cc(H'_1)) = 0$ because H'_1 is connected, and $d_{G'}(cc(H'_2)) = 3$.



components of H'_i. We define $d_{G'}(cc(H'_i)) = \max_{i,j\in\{1,\dots,s\}} d_{G'}(cc_i(H'_i), cc_j(H'_j))$. As previously said, if $|cc(H'_i)| = 1$, that is H'_i is a connected graph, then $d_{G'}(cc(H'_i)) = 0$.

Distances within G and G' shall be processed by warping functions:

Definition. 1. *(Distance warping function or DWF) A* distance warping function *is a function $g : [0, |V|] \cup \infty \to \mathbb{R}^+$, mapping distances in G to distances in G', such that:*

- *g is non-decreasing: the connectivity constraints are not stronger when the distance d increases.*
- *$g(x) \geq x$ for all $x \geq 0$: if any two nodes are at distance d in G, then we cannot constrain the distance between the two sub-graphs that receive flow from the former two nodes, to be less than d.*
- *$g(\infty) = \infty$: if two nodes $u, v \in V$ are not in a same maximal connected component in G, then any node that receives flow from u and any node that receives flow from v, can be in two different maximal connected components in G'.*

We finally arrive at connectivity constraints associated with a distance warping function:

Definition. 2. *(Connectivity constraints) Consider a distance warping function g. The connectivity constraints are satisfied for a solution f if and only if for every $i, i' \in \mathcal{I}$, then*

$$d_{G'}(H'_i, H'_{i'}) \leq g(d_G(v_i, v_{i'})) \text{ and } d_{G'}(cc(H'_i)) \leq g(0), \tag{3}$$

where H'_i ($H'_{i'}$, respectively) is the sub-graph induced by the nodes that receive flow from v_i ($v_{i'}$, respectively).

This definition calls for two comments. First, the two terms of Eq. (3) respectively define constraints associated with pairs of vertices and single vertices of the supply graph. In Lemma 1, we shall give an alternative definition of these constraints. Second, the conditions imposed with function g are analogous to Lipschitz conditions on distances. Indeed, $d_{G'}(H'_i, H'_{i'}) \leq g(d_G(v_i, v_{i'}))$ means that the distance between nodes that receive flow from v_i and nodes that receive flow from $v_{i'}$ must be upper-bounded by a function of the distance between v_i and $v_{i'}$ in G (constraints for pairs of nodes of G). Furthermore, $d_{G'}(cc(H'_i)) \leq g(0)$ means that the maximal connected components induced by the set of nodes that receive flow from v_i must be at most at distance $g(0)$ each other (constraints for single nodes of G).

As a particular case, we may ask that the two vertices defining an edge from G, and a vertex from G export flow to connected subgraphs of G'. Using the two constraints from Eq. (3), we define:

Definition. 3. *(Strict connectivity constraints)* Strict connectivity constraints *are defined by a distance warping function g such that $g(0) = 0$ and $g(x) = \infty$ for all $x \geq 1$.*

Transport size constraint. Recall that for `EMD`, the number of edges supporting flow is at most the total number of nodes minus one. Connectivity constraints may be such that a super-linear number of edges is needed. But since we do not know this number a priori, we impose an upper bound M on it. Formally, given any integer M such that $0 \leq M \leq |E(B)|$, the transport size constraint is satisfied for f if and only if $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}|f_{i,j}>0} 1 \leq M$. Practically, our implemented algorithms do not take this upper bound as input, but we carefully analyze the number of edges carrying flow and compare it against the number of nodes.

2.2 Problems `EMD-FCC` and `EMD-CCC`

Minimum-cost under flow, transport size, and connectivity constraints problem: problem `EMD-FCC`. This problem (`EMD-FCC`) consists in computing the smallest total cost when a given amount of total flow must be supported and respecting the transport size, and the connectivity constraints. Formally, given F and M, $0 \leq F \leq \min(\sum_{i\in\mathcal{I}} X_i, \sum_{j\in\mathcal{J}} Y_j)$, $0 \leq M \leq |E(B)|$, `EMD-FCC` can be written as follows:

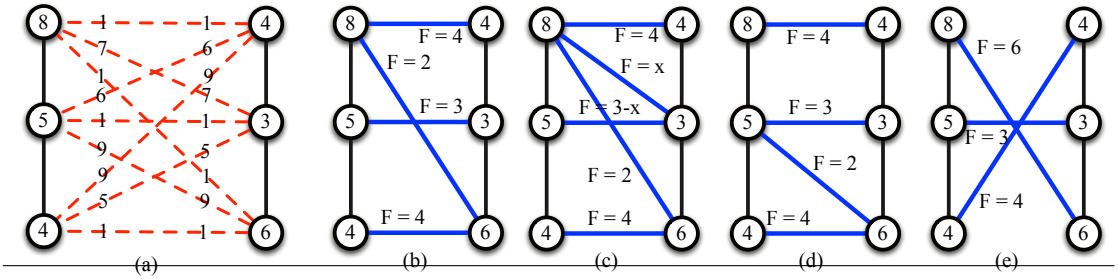
$$\begin{cases} \text{Minimize} \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}\ c_{i,j} \\ \text{subject to} \\ \sum_{i\in\mathcal{I}} f_{i,j} \leq Y_j \quad \forall j \in \mathcal{J}, \\ \sum_{j\in\mathcal{J}} f_{i,j} \leq X_i \quad \forall i \in \mathcal{I}, \\ f_{i,j} \geq 0 \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\ \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} \geq F, \\ \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}|f_{i,j}>0} 1 \leq M, \\ d_{G'}(H'_i, H'_{i'}) \leq g(d_G(v_i, v_{i'})) \quad \forall i, i' \in \mathcal{I}, \\ d_{G'}(cc(H'_i)) \leq g(0) \quad \forall i \in \mathcal{I}. \end{cases} \tag{4}$$

Line 6 represents the flow constraint, Line 7 is the transport size constraint, Lines 8 and 9 describe the connectivity constraints. Fig. 2 describes the solutions of a simple instance of `EMD-FCC`. Given an optimal solution f for `EMD-FCC`, we introduce the *total number of edges* $M_{\text{EMD-FCC}}$, the *total flow* $F_{\text{EMD-FCC}}$, the *total cost* $C_{\text{EMD-FCC}}$, and the ratio $d_{\text{EMD-FCC}}$:

$$\begin{aligned} M_{\text{EMD-FCC}} &= \sum_{i\in\mathcal{I},j\in\mathcal{J}} \mathbf{1}_{f_{i,j}>0}, F_{\text{EMD-FCC}} = \sum_{i\in\mathcal{I},j\in\mathcal{J}} f_{i,j}, \\ C_{\text{EMD-FCC}} &= \sum_{i\in\mathcal{I},j\in\mathcal{J}} f_{i,j}\ c_{i,j}, d_{\text{EMD-FCC}} = \frac{C_{\text{EMD-FCC}}}{F_{\text{EMD-FCC}}}. \end{aligned} \tag{5}$$

Maximum-flow under cost, transport size, and connectivity constraints problem: problem `EMD-CCC`. This problem (`EMD-CCC`) aims at computing the largest volume of flow that can be supported respecting the connectivity constraints, the transport size constraint and such that the total cost is less than a given bound C. We define the following upper bound $C_{max} = \sum_{j\in\mathcal{J}} Y_j \max_{i\in\mathcal{I},j\in\mathcal{J}} c_{i,j}$ for the maximum total cost of any admissible flow. Given C

Figure 2 Optimal transportation on graphs: a simple example. **(a)** The supply and demand graphs are paths; supply and demand are indicated in the nodes, while the edges are decorated with the unitary costs. **(b)** An optimal solution for `EMD`. **(c)** An admissible solution for `EMD-FCC` for any $M \in [5,9]$ and for any real number $x \in \ ]0,3]$. **(d)** An optimal solution for `EMD-FCC` for $M = 4$. **(e)** An optimal solution for `EMD-FCC` for $M = 3$. See full detail in the supplemental section 7.1.



and M, $0 \leq C \leq C_{max}$, $0 \leq M \leq |E(B)|$, `EMD-CCC` is defined as follows:

$$\begin{cases} \text{Maximize } \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} \\ \text{subject to} \\ \sum_{i\in\mathcal{I}} f_{i,j} \leq Y_j \quad \forall j \in \mathcal{J}, \\ \sum_{j\in\mathcal{J}} f_{i,j} \leq X_i \quad \forall i \in \mathcal{I}, \\ f_{i,j} \geq 0 \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \\ \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}\ c_{i,j} \leq C, \\ \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}|f_{i,j}>0} 1 \leq M, \\ d_{G'}(H'_i, H'_{i'}) \leq g(d_G(v_i, v_{i'})) \quad \forall i, i' \in \mathcal{I}, \\ d_{G'}(cc(H'_i)) \leq g(0) \quad \forall i \in \mathcal{I}. \end{cases} \tag{6}$$

Given an optimal solution f for `EMD-CCC`, we introduce the *total number of edges* $M_{\text{EMD-CCC}}$, the *total flow* $F_{\text{EMD-CCC}}$, the *total cost* $C_{\text{EMD-CCC}}$ and the ratio $d_{\text{EMD-CCC}}$:

$$\begin{aligned} M_{\text{EMD-CCC}} &= \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}|f_{i,j}>0} 1 \leq M, F_{\text{EMD-CCC}} = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}, \\ C_{\text{EMD-CCC}} &= \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j}\ c_{i,j}, d_{\text{EMD-CCC}} = \frac{C_{\text{EMD-CCC}}}{F_{\text{EMD-CCC}}}. \end{aligned} \tag{7}$$

3 The problems are very difficult to solve

We first prove that the connectivity constraints studied in this article are equivalent to constraints with general size and general number of sub-graphs of the source graph (Section 3.1). We then show the NP-completeness of `EMD-CCC` and `EMD-FCC` (Section 3.2). Furthermore, we prove a stronger hardness result with strict connectivity constraints (Section 3.3). We finally show that `EMD-FCC` is hard to approximate within any given constant (Section 3.4).

3.1 The connectivity constraints are equivalent to constraints with general size and general number of sub-graphs

We prove in Lemma 1 that the connectivity constraints studied in this article are equivalent to constraints with general size (taking sub-graphs of G of any size) and with general number of sub-graphs (taking a set $\{H_1, H_2, \ldots\}$ of sub-graphs of G of any size).

Lemma. 1. *Given any flow f, the connectivity constraints are satisfied for f if and only if $d_{G'}(H') \leq g(d_G(H))$ and $d_{G'}(cc(H'_i)) \leq g(d_G(cc(H_i)))$ for all i, $1 \leq i \leq t$, where $H = \{H_1, \ldots, H_t\}$ is any set of $t \geq 1$ disjoint sub-graphs of G, H'_i is the sub-graph of G' induced by the set of nodes that receive flow from at least one node of H_i, and $H' = \{H'_1, \ldots, H'_t\}$.*

Note in particular that under strict connectivity constraints, the previous lemma implies that the set of nodes of any connected sub-graph of G sends flow to a set of nodes of G' that induces a connected sub-graph of G'.

3.2 NP-completeness of `EMD-FCC` and `EMD-CCC`

We prove that `EMD-FCC` and `EMD-CCC` are NP-complete even for simple classes of instances. In our following reductions, we use the strongly NP-complete problem 3-Partition [7]. Let $m \geq 1$ be any integer. Given a set $S = \{n_1, n_2, \ldots, n_{3m}\}$ of $3m$ positive integers, 3-Partition problem consists in deciding if S can be partitioned into m subsets such that the sum of the numbers in each subset is equal.

Lemma. 2. *For any distance warping function g, the decision version of* `EMD-FCC` *is NP-complete even if:*

- *the demand graph G' is a complete graph;*
- *and all the volumes of demands are equal ($Y_j = Y_{j'}$ for all $j, j' \in \mathcal{J}$);*
- *and all the unitary costs are equal to one ($c_{i,j} = 1$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$);*
- *and the volumes of demands and supplies are equal ($\sum_{i \in \mathcal{I}} X_i = \sum_{j \in \mathcal{J}} Y_j$).*

We deduce in Corollary 1 that the decision version of `EMD-CCC` is NP-complete. Indeed, given a maximum cost C and a maximum number of edges M, the problem of deciding if there exists a flow f larger than a given F and satisfying all the constraints, is equivalent to the problem of deciding if there is an admissible solution for `EMD-FCC` with C as input.

Corollary. 1. *The decision version of* `EMD-CCC` *is NP-complete.*

The connectivity constraints are not directly considered in our reduction since G' is a complete graph. The choice of $M = \frac{3}{4}(|\mathcal{I}| + |\mathcal{J}|)$ is therefore the main key of the proof of Lemma 2 (and Corollary 1); this number departs from the linear number of edges (at most $|\mathcal{I}| + |\mathcal{J}| - 1$) for problem `EMD` [9]. Allowing a quadratic number of edges is covered in the next section.

3.3 Stronger NP-completeness result with strict connectivity constraints

Under strict connectivity constraints, we prove that the decision version of `EMD-FCC` and `EMD-CCC` is also NP-complete even if the upper-bound M on the number of edges that can support flow is quadratic in the number of nodes. We first establish the following result:

Lemma. 3. *Consider the strict connectivity constraints. There exists an instance of the decision version of* EMD-FCC *such that there is an admissible solution if and only if* $M \geq |\mathcal{I}|(|\mathcal{I}|+1)$ *with* $|\mathcal{J}| = 2|\mathcal{I}|$.

From which one deduces:

Lemma. 4. *Under strict connectivity constraints, the decision version of* EMD-FCC *and* EMD-CCC *is NP-complete even if* $M \in \Theta(|V \cup V'|^2)$.

3.4 Hardness of approximation of EMD-FCC

We now prove in Lemma 5 that EMD-FCC is hard to approximate. More precisely, we show that for any constant $k \geq 1$, there is no polynomial time algorithm for EMD-FCC, unless P = NP.

Lemma. 5. EMD-FCC *is not in APX even if:*

- *all the volumes of demands are equal (*$Y_j = Y_{j'}$ *for all* $j, j' \in \mathcal{J}$*);*
- *and there are only two different unitary costs for edges of the bipartite graph* B *(*$c_{i,j} \in \{1, K\}$ *for all* $i \in \mathcal{I}, j \in \mathcal{J}$*, where* $K > 1$*).*

4 Efficient polynomial time algorithms

In this section, we first describe two classes of instances that admit exact polynomial time algorithms (Section 4.1). We then prove the existence of a Polynomial Time Approximation Scheme for some classes of instances (Section 4.2). We also develop efficient polynomial time greedy algorithms (Section 4.3), and optimize its complexity for strict connectivity constraints (Section 4.4).

4.1 Exact polynomial time algorithms for some classes of instances

We prove in Lemma 6 that if the demand graph is complete or if the demand graph is connected and the distance between any two nodes is upper-bounded by the function g (for any value), then there is an exact polynomial time algorithms for EMD-FCC and EMD-CCC.

Lemma. 6. *There is a polynomial time algorithm for* EMD-FCC *and* EMD-CCC *if*

- $M = |E(B)|$ *and* G' *be a complete graph;*
- *or* $M = |E(B)|$, G' *is any connected graph, and for every* $u', v' \in V'$, $d_{G'}(u', v') \leq g(x)$ *for all* $x \geq 0$.

4.2 Polynomial Time Approximation Scheme

We prove in Lemma 7 a Polynomial Time Approximation Scheme (PTAS) for EMD-FCC when G' is a connected graph and $M = |E(B)|$. To do that, we first add a volume of flow ε for all edges of the complete bipartite graph B, and then obtain an auxiliary instance (in which we update the supply and the demand volumes for all nodes). By construction, for any distance warping function g, the connectivity constraints are satisfied, and so EMD-FCC is equivalent to EMD for this auxiliary instance, which gets solved by the linear program described in Eq. (1). Thus, we get a PTAS for EMD-FCC choosing ε function of the desired approximation ratio. An interesting problem is to determine the minimum number of edges to add in an optimal solution for EMD in order to get an admissible solution for EMD-FCC.

Lemma. 7. *Consider any distance warping function g. Let $M = |E(B)|$ and G' be any connected graph. Then, for any $\varepsilon > 0$, there is a polynomial time $(1+\varepsilon)$-approximation algorithm for* `EMD-FCC`.

In Corollary 2, we deduce the same result for `EMD-CCC`.

Corollary. 2. *Consider any distance warping function g. Let $M = |E(B)|$ and G' be any connected graph. Then, for any $\varepsilon > 0$, there is a polynomial time $(1+\varepsilon)$-approximation algorithm for* `EMD-CCC`.

4.3 Efficient polynomial time algorithms for `EMD-CCC`

Greedy algorithm. We describe here a polynomial time greedy algorithm for `EMD-CCC` (Algorithm 1). The inputs are the graphs $G = (V, E)$ and $G' = (V', E')$, a cost upper bound C, a maximum number M of edges that can support flow, and a distance warping function g for the connectivity constraints. Algorithm 1 greedily selects edges of the bipartite graph that can support flow such that the total cost is upper bounded by C, and without violating the transport size and the connectivity constraints. After such a selection, the set of candidate edges for the next step of selection is updated with respect to the connectivity constraints. Algorithm 1 returns the solution f, the total flow F_f, the total cost C_f, and the total number of edges M_f.

Algorithm 1 Greedy algorithm for `EMD-CCC`.

Require: $G = (V, E)$, $G' = (V', E')$, C, M, g.
1: $F_f := 0$; $C_f := 0$; $M_f := 0$;
2: **for** all $i \in \mathcal{I}, j \in \mathcal{J}$ **do**
3: $\quad f_{i,j} := 0$; $x_i := X_i$; $y_j := Y_j$; $b_{i,j} := 1$
4: **while** $C_f < C$, $M_F \leq M - 1$, and $\exists (i,j)$ such that $b_{i,j} = 1$, $x_i > 0$, and $y_j > 0$ **do**
5: $\quad (i_t, j_t) = \underset{i \in \mathcal{I}, x_i > 0, j \in \mathcal{J}, y_j > 0}{\arg\min} c_{i,j} b_{i,j}$; $z := \min(\frac{C - C_f}{c_{i_t,j_t}}, \min(x_{i_t}, y_{j_t}))$
6: $\quad F_f := F_f + z$; $C_f := C_f + z.c_{i_t,j_t}$; $M_F := M_F + 1$;
7: $\quad f_{i_t,j_t} := f_{i_t,j_t} + z$; $x_{i_t} := x_{i_t} - z$; $y_{j_t} := y_{j_t} - z$;
8: $\quad$ Update of $b_{i,j}$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$
9: **return** f, F_f, C_f, M_f.

Variables of Algorithm 1. For all $i \in \mathcal{I}$, $x_i \geq 0$ represents the current volume of supply of node $v_i \in V$, and so $X_i - x_i$ is the current amount of flow sent by v_i. For all $j \in \mathcal{J}$, $y_j \geq 0$ represents the current volume of demand of node $v'_j \in V'$, and so $Y_j - y_j$ is the current amount of flow received by v'_j. For all $v_i \in V, v'_j \in V'$, the variable b_{v_i,v'_j} is used to encode if the edge $\{v_i, v'_j\} \in E(B)$ can support flow in respect with the constraints. When there is no ambiguity, we abuse the notation writing $b_{i,j}$ instead of b_{v_i,v'_j}. In other words, $b_{i,j} = 1$ if the edge $\{v_i, v'_j\}$ is an edge candidate ($b_{i,j} = 0$ otherwise). The variable f represents the current solution. In other words, for all $i \in \mathcal{I}, j \in \mathcal{J}$, $f_{i,j}$ represents the current flow sent from $v_i \in V$ to $v'_j \in V'$. Furthermore, F_f is the total volume of the current flow, C_f represents the total cost of the current flow, M_f is the current number of edges of B that support flow, and F_f is the total volume of the current flow. Initially, $F_f = 0$, $C_f = 0$, $M_f = 0$, $f_{i,j} = 0$, $x_i = X_i$, $y_j = Y_j$, and $b_{i,j} = 1$ for all $i \in \mathcal{I}, j \in \mathcal{J}$.

Core of Algorithm 1. While the current cost C_f is less than the given cost upper bound C, while the current number M_f of edges of B that support flow is strictly less than the given upper bound M, and while there exists an edge candidate $\{v_i, v'_j\}$ such that $x_i, y_j > 0$ (that is

such that a positive flow can be supported by $\{v_i, v'_j\} \in E(B)$), then an edge $\{v_{i_t}, v'_{j_t}\} \in E(B)$ is selected (Line 5). Then, the maximum amount of flow z that can be supported by the edge $\{v_{i_t}, v'_{j_t}\}$ is computed. Line 6 updates the values of C_f, M_f, and F_f. Line 7 updates the values of f_{i_t,j_t}, x_{i_t}, and y_{j_t}. Line 8 updates the boolean function b for all $i \in \mathcal{I}, j \in \mathcal{J}$. Algorithm 1 finally returns f, F_f, C_f, and M_f (Line 9).

Time complexity of Algorithm 1. The time complexity of our greedy algorithm is a function of the time complexity of updating $b_{i,j}$, and so function of the connectivity constraints. We formalize that in Lemma 8.

Lemma. 8. *The time complexity of Algorithm 1 is* $O(\max(|V|^2|V'|^2, |V||V'|C(g)))$*, where* $C(g)$ *is the time complexity of updating* b *with* g *representing the connectivity constraints.*

We propose in Algorithm 2 a polynomial time algorithm for updating the boolean function b used in Algorithm 1.

Algorithm 2 Update of the boolean function b used in Algorithm 1 for general connectivity constraints.

Require: $G = (V, E)$, $G' = (V', E')$, (i_t, j_t), b, f, x.
Ensure: Binary values $b_{i,j}$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$.

1: **for** all $i \in \mathcal{I}$ **do**
2: **for** all $j \in \mathcal{J}$ **do**
3: $b'_{i,j} := 1$
4: **for** all $i \in \mathcal{I}$ **do**
5: **for** all $j \in \mathcal{J}$ **do**
6: **if** $d_{G'}(cc(G'[V(H'_i) \cup \{v'_j\}])) > g(0)$ **then**
7: $b'_{i,j} := 0$
8: **for** all $i_1 \in \mathcal{I}$ **do**
9: **for** all $i_2 \in \mathcal{I}$ **do**
10: **for** all $j \in \mathcal{J}$ **do**
11: **if** $d_{G'}(G'[V(H'_{i_1}) \cup \{v'_j\}], H'_{i_2}) > g(d_G(v_{i_1}, v_{i_2}))$ **then**
12: $b'_{i_1,j} := 0$
13: **if** $d_{G'}(H'_{i_1}, G'[V(H'_{i_2}) \cup \{v'_j\}]) > g(d_G(v_{i_1}, v_{i_2}))$ **then**
14: $b'_{i_2,j} := 0$
15: **for** all $i \in \mathcal{I}$ **do**
16: **for** all $j \in \mathcal{J}$ **do**
17: $b_{i,j} := b'_{i,j}$
18: **return** b.

The following directly follows from the definition of the connectivity constraints:

Lemma. 9. *Algorithm 2 updates* $b_{i,j}$ *for all* $i \in \mathcal{I}, j \in \mathcal{J}$ *for general connectivity constraints.*

Lemma. 10. *The time complexity of Algorithm 2 is* $O(\max(|V|^3, |V|^2|V'|^3 \log(|V'|)))$.

From Lemma 8 and Lemma 10, we deduce in Corollary 3 the time complexity of Algorithm 1 when using Algorithm 2 for the update of b.

Corollary. 3. *The worst case time-complexity of Algorithm 1 with Algorithm 2 for the update of* b*, is* $O(\max(|V|^4|V'|, |V|^3|V'|^4 \log(|V'|)))$.

Remark 1. *Selecting a single edge per step in Algorithm 1 calls for two comments. First, the selection of a* best *set of* $k > 1$ *edges at each, does not necessarily guarantee a better solution at*

the end of the algorithm. Indeed, we can easily construct classes of instances for which such an algorithm (that selects several edges per step) is less efficient than the algorithm presented before (in terms of volume of flow and cost). Second, such a multiple edges selection would increase the time complexity of the update of the boolean function b and so of the greedy algorithm. For these two reasons, we decide to focus on algorithms that select single edge per step.

Iterative algorithm. The maximum cost is an input of Algorithm 1. Since we do not know, a priori, the cost of interesting flow solutions, we must call Algorithm 1 with different input costs, each call generating different statistics as defined by Eq. (7). To this end, given a cost range $[0, C_{max}]$, we now describe the iterative algorithm Algorithm 3 which computes different flow solutions for different total costs, by iteratively calling Algorithm 1. Note that practically, C_{max} can be set to the maximum distance between a supply and a demand node, times the total supply.

Algorithm 3 Iterative algorithm for `EMD-CCC`.

Require: $G = (V, E)$, $G' = (V', E')$, C_{inf}, C_{sup}, g.
1: F_{inf} := total flow returned by Algorithm 1 with G, G', $C := C_{inf}$, $M := |E(B)|$, g
2: F_{sup} := total flow returned by Algorithm 1 with G, G', $C := C_{sup}$, $M := |E(B)|$, g
3: **if** $F_{inf} < F_{sup}$ **then**
4: Algorithm 3 with G, G', C_{inf}, $C_{sup} := C$, g
5: Algorithm 3 with G, G', $C_{inf} := C$, C_{sup}, g

4.4 Results for strict connectivity constraints

In the sequel, we describe Algorithm 4, a polynomial time algorithm improving the generic update of the boolean function b (algorithm 2) for the particular case of strict connectivity constraints (Def. 3).

We use the following notations. For any subset $S \subseteq V$, the open neighborhood $N_G(S)$ of S is the set of nodes in $V \setminus S$ having a neighbor in S and the closed neighborhood of S, denoted by $N_G[S]$, is defined as $N(S) \cup S$. If $S = \{v\}$, we use $N_G(v)$ and $N_G[v]$ instead of $N_G(\{v\})$ and $N_G[\{v\}]$, respectively. We denote by $\Delta(G)$ the maximum degree of G.

The following lemma proves that Algorithm 4 updates the set of candidate edges that can support flow under strict connectivity constraints:

Lemma. 11. *Algorithm 4 updates $b_{i,j}$ for all $i \in \mathcal{I}, j \in \mathcal{J}$ for the strict connectivity constraints.*

The corresponding complexity satisfies:

Lemma. 12. *The time complexity of Algorithm 4 is $O(|V'| + \Delta(G)^2\Delta(G'))$.*

From Lemma 8 and Lemma 12, we deduce in Corollary 4 the time complexity of Algorithm 1 when using Algorithm 4 for the update of b with strict connectivity constraints.

Corollary. 4. *The worst case time-complexity of Algorithm 1 with Algorithm 4 for the update of b, is $O(\max(|V|^2|V'|^2, \Delta(G)^2\Delta(G')|V||V'|))$.*

This result shows the decrease of the time complexity when we consider the strict connectivity constraints. Indeed, for general connectivity constraints, recall that the worst case time-complexity of Algorithm 1 is $O(\max(|V|^4|V'|, |V|^3|V'|^4 \log(|V'|)))$ (Corollary 3).

Algorithm 4 Update of the boolean function b used in Algorithm 1 for the strict connectivity constraints.

Require: $G = (V, E)$, $G' = (V', E')$, (i_t, j_t), b, f, x.
Ensure: Binary values $b_{i,j}$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$.

if $X_{i_t} - x_{i_t} - f_{i_t,j_t} = 0$ **then**
  for all j such that $v'_j \in N_{G'}(v'_{j_t})$ **do** $b_{i_t,j} := 1$
  for all j such that $v'_j \notin N_{G'}(v'_{j_t})$ **do** $b_{i_t,j} := 0$
  for all i such that $v_i \in N_G(v_{i_t})$ **do**
    if $X_i - x_i = 0$ **then**
      for all j such that $v'_j \notin N_{G'}[v'_{j_t}]$ **do** $b_{i,j} := 0$
else
  for all j such that $v'_j \in N_{G'}(v'_{j_t})$ **do** $b_{i_t,j} := 1$
  for all i such that $v_i \in N_G(v_{i_t})$ **do**
    if $X_i - x_i = 0$ **then**
      for all j such that $v'_j \in N_{G'}[v'_{j_t}]$ **do**
        if $b_{i,j} = 0$ **then**
          $b_{i,j} := 1$
          for all k such that $v_k \in N_G(v_i)$ **do**
            if $X_k - x_k > 0$ and $b_{k,j} = 0$ **do** $b_{i,j} := 0$
return b.

Note that the maximum degrees of G and G' may be linear in the number of nodes. In that case, the time complexity is $O(|V|^3|V'|^2)$ but, in general, the time complexity is better. For instance, Corollary 5 proves a better complexity if G and G' have bounded degrees. More precisely, the complexity only depends on Algorithm 1.

Corollary. 5. *The worst case time-complexity of Algorithm 1 with Algorithm 4 for the update of b, is* $O(|V|^2|V'|^2)$ *if* $\Delta(G) = O(1)$ *and* $\Delta(G') = O(1)$.

5 Experiments

We benchmark our implementations `Alg-EMD-CCC-G` (Algorithm 1 solving `EMD-CCC`) and `Alg-EMD-LP` (solving `EMD`) to compare clusterings and analyze molecular data.

5.1 Implementations

The Structural Bioinformatics Library is a generic C++/python cross-platform software library targeting complex problems in structural bioinformatics [2]. It combines low-level generic (C++ template based) implementations of various algorithms–in a spirit analogous to CGAL–defining `SBL-CORE`, and their instantiations to solve specific biophysical problems–defining `SBL-APPLICATIONS`. Low level generic C++ implementations of our algorithms are available in the following package from `SBL-CORE` (`http://sbl.inria.fr/doc/group__Earth__mover__distance-package.html`); instantiations of these methods for PEL are available from the following package from `SBL-APPLICATIONS` `http://sbl.inria.fr/doc/Energy_landscape_comparison-user-manual.html`.

5.2 Comparing clusterings

Setup. Clustering is key in data analysis, yet, the variety of options (algorithms and their parameters) makes comparing clusterings a key endeavor. As an illustration, we use clusterings computed by mode-seeking methods [4]. These methods identify clusters from the catchment basins of (persistent) local maxima of an estimated density. They inherently embed the notion of persistence for clusters, providing insights on the number of clusters–as opposed to requiring a pre-defined number of clusters. Finally, edges of the clustering graph are naturally defined by the saddles associated to the local maxima defining the clusters [4]. Summarizing, we consider *clustering graphs* (CG) defined as follows:

- Vertices. There is one node per cluster. The cluster is endowed with a representative point of that cluster (e.g. its centroid), used to compute the unit costs c_{ij}. The weight of a node i.e. the supply or demand is the number of input points in that cluster.

- Edges. Selected pairs of clusters are connected, based on a predicate (clustering algorithm dependent).

To compare two clusterings, we enforce strict connectivity constraints, mapping connected regions associated with the first CG to connected regions associated with the second CG.

Data. We compare clusterings of data points drawn from a mixture of 5 anisotropic gaussians. More precisely, each gaussian is defined by its center and the rotation angle α (in degrees) of its principal direction, yielding a triple (c_x, c_y, α). The five gaussian used have parameters $(-d, d, 45), (-d, -d, -45), (0, d/2., 0), (2{*}d, 2{*}d, 135), (2{*}d, 2{*}d, 45)$. Using $d = \{0, 40, 50\}$ yields three mixtures. Clustering $N = 5000$ points drawn at random yields three clusterings, whence three pairwise comparisons.

Results. For a given pair, `Alg-EMD-CCC-G` is run twice since it is not symmetric. As for `Alg-EMD-LP`, the demand is always satisfied; but we monitor the fraction of vertices ($r_V^{\mathrm{c.c.}}$) and edges ($r_E^{\mathrm{c.c.}}$) satisfying the strict connectivity constraints.

While $d_{\text{EMD-CCC}} < d_{\text{EMD}}$ always holds, `Alg-EMD-CCC-G` falls short from satisfying the demand (% flow in the range 13-32%). On the other hand, `Alg-EMD-LP` fails from satisfying connectivity constraints, in particular for edges ($r_E^{\mathrm{c.c.}} \in 0.24 - 0.45\%$). See Table 1.

Figure 3 Comparing clustering of 5000 points drawn according to a mixture of 5 gaussians. From left to right: $d = 50$; middle: $d = 40$; right $d = 0$ (see text for details). Algorithm `Tomato` [4] yielded 44, 38 and 41 clusters, respectively, using a persistence threshold of $7.5 * 10^{-3}$.

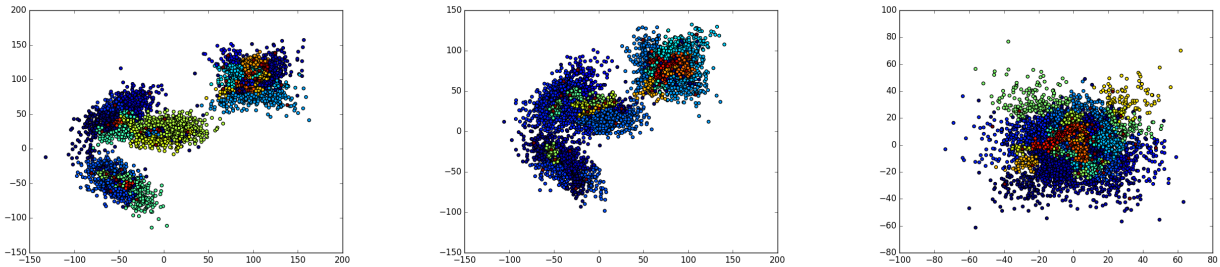
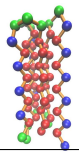


Table 1 Comparisons for the 3 clusterings of Fig.3: performances. The 3 rows correspond to the three pairwise comparisons. Top table: `Alg-EMD-CCC-G` (two calculations since the algorithm is not symmetric) Bottom table: `Alg-EMD-LP`

	$d=0$ vs $d=40$	$d=0$ vs $d=50$	$d=40$ vs $d=50$
$d_{\text{EMD-CCC}}$ (%flow) for (A,B)	61.78 (24%)	55.57 (32%)	12.11 (13%)
$d_{\text{EMD-CCC}}$ (%flow) for (B,A)	25.47 (32%)	37.10 (31%)	22.76 (25%)
d_{EMD}	65.09	80.96	26.66
$r_V^{\text{c.c.}}$	0.94 - 0.94	0.93 - 0.94	0.95 - 0.97
$r_E^{\text{c.c.}}$	0.24 - 0.29	0.24 - 0.28	0.45 - 0.48

Figure 4 Conformation of the BLN69 model.



5.3 Analysis of molecular energy landscapes

Setup. The potential energy landscape (PEL) of a molecular system is the graph of the function associating a potential energy to each conformation [18]. PEL codes all thermodynamic and kinetic properties, and of particular importance are local minima and the (index one) saddles connecting them. Because PEL usually exhibits a number of minima exponential in the number of degree of freedom ($3n$ of them for n atoms, e.g. $n = 5,000$ for a medium sized protein), a pruning of local minima is generally in order. Topological persistence proved instrumental in this respect [3, 1]. Summarizing, we consider so-called *transition graphs (TG)*:

- Vertices. One vertex for each persistent local minimum. The unit cost c_{ij} between two vertices is taken as the *root mean square deviation* between the two conformations associated with the local minima. The weight (supply or demand) is theoretically defined by the integral of Boltzmann's factor over is catchment basin. Practically, using thermodynamic ensembles, it is estimated from the number of points in the basin.

- Edges. Two vertices sharing an index one saddle on the PEL.

We use our algorithms to compare two TG generated by two Monte Carlo runs. The comparisons aim at assessing the coherence between two explorations of the PEL, enforcing strict connectivity constraints to map connected regions of the two TG with one another.

Practically, we use BLN69, a model protein with three types of pseudo amino-acids (a.a.) namely hydrophobic (B), hydrophylic (L) and neutral (N) [13]. This system has 69 a.a., whence a total of 207 Cartesian coordinates. The corresponding PEL has been thoroughly studied, with of the order of one million of local minima reported by state-of-the-art Monte Carlo methods [13, 14], and 10 low lying local minima identified.

We define a *sampling* as $N = 10^4$ conformations generated by `T-RRT` [10], a randomized incremental algorithm favoring the exploration of *not-visited-yet* regions. We further quench each sample p_i to its local minimum $q(p_i)$, by performing a gradient descent on the PEL. To convert the sampling into a TG, we use the `Tomato` algorithm [4]. This algorithm, which is reminiscent of clustering by mode seeking, identifies *saddles* at which the *basins* of nearby local minima merge.

The resulting graph is called the *transition graph* (TG). Furthermore, we simplify the TG using topological persistence, to retain the 50 most significant local minima. For this simplified TG, the mass of each basin is the fraction of samples located in its basin. This operation was carried out once for each local minimum in the top ten. We refer to this data set as TRRT-top10, and to a particular graph as TRRT-top10-i, $i = 1, \ldots, 10$.

Our ten TG yield 45 pairs, whence 45 instances for `Alg-EMD-LP` (which is symmetric), and 90 instances for `Alg-EMD-CCC-G`.

Results: algorithm `Alg-EMD-LP` and constraint satisfaction. Since `Alg-EMD-LP` is oblivious to critical point connectivity, we compute the fraction $r_V^{\text{c.c.}}$ and $r_E^{\text{c.c.}}$ of vertices and edges of the input graph inducing through the flow a connected subgraph of the demand graph. Out of the 45 instances of the dataset TRRT-top10, the min, median and max values for vertices and edges are (0.1, 0.62, 1.), and (0.03, 0.89, 1.), respectively. That is, transport plans obtained from solutions of the linear program do disrupt connectivity constraints.

Results: algorithm `Alg-EMD-CCC-G` and demand satisfaction. The connectivity constraints may prevent `Alg-EMD-CCC-G` to fully satisfy the demand. For each instance, we therefore monitor the total flow $F_{\text{Alg-EMD-CCC-G}}$ provided by the transport plan, the ideal value being one. On these 90 instances, a worst-case of 0.99 is observed. Further inspection shows that such performances owe to the distribution of weights in the basins. Indeed, for each transition graph TRRT-top10-i, it turns out that the local minimum from which the exploration was started takes most of the mass. Therefore, in comparing two such graphs, a transportation plan essentially reduces to moving the mass in-between the two prominent local minima. For this particular application, the fact that connectivity constraints are lenient shows that all runs discovered the same local minima and transitions. This stability is informative and gives confidence for physical analysis carried out downstream.

Results: transport costs. To assess transport costs, we compute the linear correlation between three sets of 45 values, namely the transport costs of `Alg-EMD-LP` of the 45 instances, and those of `Alg-EMD-CCC-G` on the 45×2 pairs (recall that `Alg-EMD-CCC-G` is not symmetric). The three coefficients obtained are equal to 0.999, a property again owing to the structure of the basins, as just discussed.

6 Conclusion

This paper introduces optimal transportation problems which depart from classical ones as they embed connectivity constraints, and shows that these problems are in general hard to solve. A greedy polynomial time algorithm is also proposed for one of them, which is of particular interest to compare graphs representing clusterings and molecular energy landscapes. Our experiments show that optimizing the transport plan and respecting connectivity constraint can be competing objectives. On the other hand, comparable transport plans observed without and with connectivity constraints show that the supply and demand graphs are *comparable*, both in terms of embedding of vertices and edges. This information is especially interesting for selected applications, in particular (molecular) simulation, as it shows the stability of the data generation processes.

On the theoretical side, future work will aim at understanding whether problems with cost and connectivity constraints are hard to approximate, to possibly develop efficient algorithms. On the applied side, the importance of distance warping functions will be studied in the context of our two applications (comparing clusterings, analyzing molecular data).

References

[1] J. Carr, D. Mazauric, F. Cazals, and D. J. Wales. Energy landscapes and persistent minima. *The Journal of Chemical Physics*, 144(5), 2016.

[2] F. Cazals and T. Dreyfus. The Structural Bioinformatics Library: modeling in biomolecular science and beyond. *Bioinformatics*, NA(NA), 2016.

[3] F. Cazals, T. Dreyfus, D. Mazauric, A. Roth, and C.H. Robert. Conformational ensembles and sampled energy landscapes: Analysis and comparison. *J. of Computational Chemistry*, 36(16):1213–1231, 2015.

[4] F. Chazal, L. Guibas, S. Oudot, and P. Skraba. Persistence-based clustering in riemannian manifolds. *J. ACM*, 60(6):1–38, 2013.

[5] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE PAMI*, 17(8):790–799, 1995.

[6] G. B. Dantzig. Application of the simplex method to a transportation problem. *Activity Analysis of Production and Allocation*, pages 359–373, 1951.

[7] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[8] A. Genevay, M. Cuturi, G. Peyré, and F. Bach. Stochastic optimization for large-scale optimal transport. In *Proc. NIPS'16*, 2016.

[9] F. Hillier and G. Lieberman. *Introduction to mathematical programming*. McGraw-Hill, 1977.

[10] L. Jaillet, F.J. Corcho, J-J. Pérez, and J. Cortés. Randomized tree construction algorithm to explore energy landscapes. *Journal of computational chemistry*, 32(16):3464–3474, 2011.

[11] E. Levina and P. Bickel. The earth mover's distance is the mallows distance: Some insights from statistics. In *IEEE ICCV*, volume 2, pages 251–256. IEEE, 2001.

[12] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences de Paris*, pages 666–704, 1781.

[13] M.T. Oakley, D.J. Wales, and R.L. Johnston. Energy landscape and global optimization for a frustrated model protein. *The J. of Phys. Chem. B*, 115(39):11525–11529, 2011.

[14] A. Roth, T. Dreyfus, C.H. Robert, and F. Cazals. Hybridizing rapidly growing random trees and basin hopping yields an improved exploration of energy landscapes. *J. of Computational Chemistry*, 37(8):739–752, 2016.

[15] Y. Rubner, C. Tomasi, and L.J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[16] J. Solomon, F. De Goes, G. Peyré, M. Cuturi, A. Butscher, A. Nguyen, T. Du, and L. Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics (TOG)*, 34(4):66, 2015.

[17] C. Villani. *Topics in optimal transportation*. Number 58. AMS, 2003.

[18] D. J. Wales. *Energy Landscapes*. Cambridge University Press, 2003.

7 Appendix

7.1 A simple example with strict connectivity constraints

In the following, we detail the example of Fig. 2:

- Fig. 2 (a) describes a simple instance. Let $V = \{v_1, v_2, v_3\}$ be a set of three supply nodes such that $X_1 = 8$, $X_2 = 5$, $X_3 = 4$. Let $V' = \{v'_1, v'_2, v'_3\}$ be a set of three demand nodes such that $Y_1 = 4$, $Y_2 = 3$, $Y_3 = 6$. Integers on nodes represent these supply and demand values. The graph $G = (V, E)$ is a path, where $E = \{\{v_1, v_2\}, \{v_2, v_3\}\}$. The graph $G' = (V', E')$ is also a path, where $E' = \{\{v'_1, v'_2\}, \{v'_2, v'_3\}\}$. Integers on edges of the complete bipartite graph B represent unitary costs $c_{i,j}$ for all $i, j \in \{1, 2, 3\}$. The unitary costs are $c_{1,1} = 1$, $c_{1,2} = 7$, $c_{1,3} = 1$, $c_{2,1} = 6$, $c_{2,2} = 1$, $c_{2,3} = 9$, $c_{3,1} = 9$, $c_{3,2} = 5$, and $c_{3,3} = 1$.

- Fig. 2 (b) represents an optimal solution for `EMD`: $f_{1,1} = 4$, $f_{1,3} = 2$, $f_{2,2} = 3$, $f_{3,3} = 4$, and $f_{1,2} = f_{2,1} = f_{2,3} = f_{3,1} = f_{3,2} = 0$. Only links of cost 1 are used and so the cost of the solution is $C_{\text{EMD}} = \sum_{j \in \{1,2,3\}} Y_j = 13$. This solution is not admissible for `EMD-FCC`. Indeed node $v_1 \in V$ sends flow only to demand nodes $v'_1 \in V'$ and $v'_3 \in V'$ (that is $f_{1,1} > 0$, $f_{1,2} = 0$, and $f_{1,3} > 0$), and the nodes v'_1 and v'_3 do not induce a connected sub-graph because $\{v'_1, v'_3\} \notin E'$. One can observe that there does not exist an admissible solution of cost 13 for `EMD-FCC` even when $M = |E(B)| = 9$. In the following, we consider `EMD-FCC` with $F = \sum_{j \in \mathcal{J}} Y_j = 13$.

- Fig. 2 (c) represents an admissible solution for `EMD-FCC` for any $M \in [5, 9]$, and for any real number $x \in \ ]0, 3]$: $f_{1,1} = 4$, $f_{1,2} = x$, $f_{1,3} = 2$, $f_{2,2} = 3 - x$, $f_{3,3} = 4$, and $f_{2,1} = f_{2,3} = f_{3,1} = f_{3,2} = 0$. The total cost is $C_{\text{EMD-FCC}} = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} = 6x + 13$. Thus, $\lim_{x \to 0} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} = 13$ but we cannot obtain an admissible solution of cost 13 because $x > 0$.

- Fig. 2 (d) shows an optimal solution for `EMD-FCC` for $M = 4$: $f_{1,1} = 4$, $f_{2,2} = 3$, $f_{2,3} = 2$, $f_{3,3} = 4$, and $f_{1,2} = f_{1,3} = f_{2,1} = f_{3,1} = f_{3,2} = 0$. The total cost is $C_{\text{EMD-FCC}} = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} = 29$.

- Fig. 2 (e) describes an optimal solution for `EMD-FCC` for $M = 3$: $f_{1,3} = 6$, $f_{2,2} = 3$, $f_{3,1} = 4$, and $f_{1,1} = f_{1,2} = f_{2,1} = f_{2,3} = f_{3,2} = f_{3,3} = 0$. The total cost is $C_{\text{EMD-FCC}} = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} = 45$. One can observe that there does not exist an admissible solution for `EMD-FCC` when $0 \leq M \leq 2$.

7.2 Proofs for hardness – section 3

Proof of Lemma 1. We prove that the connectivity constraints are satisfied (for every H) if and only if these constraints are satisfied for $H = \{H_1, H_2\}$ such that $|H_1| = |H_2| = 1$ and $H_1 \neq H_2$. In the following, we set $H_1 = \{v_a\}$ and $H_2 = \{v_b\}$ such that $v_a \neq v_b$.

($\Rightarrow$) If the connectivity constraints are satisfied, then, clearly, the connectivity constraints are satisfied for every $H = \{H_1, H_2\}$ such that $H_1 = \{v_a\}$ and $H_2 = \{v_b\}$, $v_a \neq v_b$.

($\Leftarrow$) Suppose that the connectivity constraints are not satisfied for some $H = \{H_1, \ldots, H_t\}$, $t \geq 1$. Let $H' = \{H'_1, \ldots, H'_t\}$, where H'_i is the set of nodes of G' that receive flow from at least one node of H_i, for all i, $1 \leq i \leq t$. We prove that there exists two nodes $v_1 \in V(G)$, and $v_2 \in V(G)$, $v_1 \neq v_2$ such that either $d_{G'}(H'_{v_1}, H'_{v_2}) > g(d_G(v_1, v_2))$, or $d_{G'}(cc(H'_{v_1})) > g(0)$, or $d_{G'}(cc(H'_{v_2})) > g(0)$, where H'_{v_i} is the sub-graph induced by the set of nodes that receive flow

from v_i, $i \in \{1,2\}$. Without loss of generality, we have denoted such two nodes v_1 and v_2. There are two cases:

- First, suppose that the connectivity constraints are not satisfied because $d_{G'}(H') > g(d_G(H))$. It means that there exist i,j, $1 \leq i < j \leq t$, such that $d_{G'}(H'_i, H'_j) > g(d_G(H))$. Observe that $g(d_G(H)) \geq g(d_G(H_i, H_j))$. Furthermore, for every two nodes $v_1 \in V(H_i)$, and $v_2 \in V(H_j)$, $v_1 \neq v_2$, then $d_G(v_1, v_2) \leq d_G(H_i, H_j)$. Observe also that $H'_{v_1} \subseteq H'_i$ and $H'_{v_2} \subseteq H'_j$ because $v_1 \in H_1$ and $v_2 \in H_2$. Indeed a node of G' that receive flow from v_1 (resp. v_2), receive flow from at least one node of H_i (resp. H_j). Thus, $d_{G'}(H'_{v_1}, H'_{v_2}) \geq d_{G'}(H'_i, H'_j)$. Recall that it is assumed that $d_{G'}(H'_i, H'_j) > g(d_G(H))$. Then, it follows that $d_{G'}(H'_{v_1}, H'_{v_2}) > g(d_G(H))$. Furthermore, since we have proved before that $g(d_G(H)) \geq g(d_G(H_i, H_j))$ and $d_G(v_1, v_2) \leq d_G(H_i, H_j)$, then $g(d_G(H)) \geq g(d_G(v_1, v_2))$ because g is non-decreasing. Finally, we get that $d_{G'}(H'_{v_1}, H'_{v_2}) > g(d_G(v_1, v_2))$.

- Second, suppose that the connectivity constraints are not satisfied because there exists i, $1 \leq i \leq t$, such that $d_{G'}(cc(H'_i)) > g(d_G(cc(H_i)))$. There exist j, j', $1 \leq j < j' \leq s'$, such that $d_{G'}(cc_j(H'_i), cc_{j'}(H'_i)) > g(d_G(cc(H_i)))$. Suppose first that there exist two nodes $u, v \in V(G)$ such that $u \in cc_k(H_i)$, $v \in cc_{k'}(H_i)$ with $1 \leq k < k' \leq s = |cc(H_i)|$, and such that both u and v send flow to at least one node of $cc_j(H'_i)$ and at least one node of $cc_{j'}(H'_i)$. Then, $d_{G'}(cc_l(H'_u), cc_{l'}(H'_v)) > g(d_G(u, v))$, where $cc_l(H'_u)$ (resp, $cc_{l'}(H'_v)$) is a maximal connected component of the sub-graph induced by the set of nodes that receive a flow from at least u or v that is contained in $cc_j(H'_i)$ (resp. $cc_j(H'_i)$). Indeed, $g(d_G(u, v)) \geq g(d_G(cc(H_i))$ because g is non-decreasing and $d_{G'}(cc_l(H'_u), cc_{l'}(H'_v)) > d_{G'}(cc_j(H'_i), cc_{j'}(H'_i))$. Thus, it is done for this case (the connectivity constraints are not satisfied when choosing these two nodes). Then, suppose that such two nodes do not exist. Thus, it necessarily means that there exists one node $v \in V(H_i)$ such that v sends flow to a set $H'_v \subseteq H'_i$ such that $d_{G'}(cc(H'_v)) > g(d_G(\{v\}))) = g(0)$. Thus the connectivity constraints are not satisfied for one node.

We have proved that the connectivity constraints are satisfied (for every H) if and only if these constraints are satisfied for every $H = \{H_1, H_2\}$ such that $|H_1| = |H_2| = 1$ and $H_1 \neq H_2$. □

Proof of Lemma 2. Consider an instance of 3-Partition problem. Let $m \geq 1$ be any integer and let $S = \{n_1, n_2, \ldots, n_{3m}\}$ be a set of $3m$ positive integers. We construct the instance of `EMD-FCC` as follows. Set $|\mathcal{I}| = 3m$ and $|\mathcal{J}| = m$. Set $X_i = n_i$ for all $i \in \mathcal{I}$. Let $Z = \sum_{i \in \mathcal{I}} X_i$. Without loss of generality, let $Y_j = Y$ with $Z = mY$. Set $c_{i,j} = 1$ for all $i \in \mathcal{I}, j \in \mathcal{J}$. Let $G = (V, E)$ be any connected graph and let $G' = (V', V' \times V')$. Let $F = Z$, $M = 3m$, and $D = m$. Since G' is a complete graph, the connectivity constraints are always satisfied (for any function g). We prove that there is an admissible solution for `EMD-FCC` if and only if there is a solution for the instance of 3-Partition problem.

($\Leftarrow$) Assume there is a solution for the instance of 3-Partition problem, that is S can be partitioned into m subsets $S_1, S_2, \ldots, S_m$ such that the sum of the numbers in each subset is equal. We construct our solution for `EMD-FCC` as follows. For all $i \in \mathcal{I}, j \in \mathcal{J}$, if $n_i \in S_j$, then set $f_{i,j} := n_i$, otherwise set $f_{i,j} := 0$. By construction, we have $0 \leq f_i \leq X_i$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$. Since $S_1, S_2, \ldots, S_m$ is a solution for the instance of 3-Partition problem, then $\sum_{i \in \mathcal{I}} f_{i,j} = Y_j$ for all $j \in \mathcal{J}$. Finally we prove that the number of edges of B that support flow is (at most) $M = 3m$. By construction, $f_{i,j_1} f_{i,j_2} = 0$ for all $i \in \mathcal{I}$, $j_1, j_2 \in \mathcal{J}$. Thus, for all $i \in \mathcal{I}$, there is at most one edge adjacent to v_i that supports flow. Thus, the solution is admissible because $|\mathcal{I}| = 3m$.

($\Rightarrow$) Assume there is an admissible solution for `EMD-FCC`. Since $\sum_{i \in \mathcal{I}} X_i = \sum_{j \in \mathcal{J}} Y_j$, then $\sum_{j \in \mathcal{J}} f_{i,j} > 0$ for all $i \in \mathcal{I}$. In other words, there is at least one edge adjacent to v_i that supports

flow for all $i \in \mathcal{I}$. Furthermore there is at most one edge adjacent to v_i that supports flow for all $i \in \mathcal{I}$ because $M = 3m = |\mathcal{I}|$. Thus, for all $i \in \mathcal{I}$, there is exactly one edge adjacent to v_i that supports flow. We construct a solution for the instance of 3-Partition problem as follows. For all $i \in \mathcal{I}, j \in \mathcal{J}$, if $f_{i,j} > 0$, then $n_i \in S_j$, otherwise set $n_i \notin S_j$. By hypothesis (existence of an admissible flow), the sum of the numbers of S_j is Y because $\sum_{i\in\mathcal{I}} X_i = \sum_{j\in\mathcal{J}} Y$. Thus, there is a solution for the instance of 3-Partition problem.

In conclusion, the decision version of `EMD-FCC` is strongly NP-complete because it is in NP and 3-Partition problem is strongly NP-complete [7]. □

Proof of Lemma 3. Let $t \geq 1$ be any integer. Set $|\mathcal{I}| = t+1$ and set $|\mathcal{J}| = 2(t+1)$. Let $a \geq t$ be any real number and let $\varepsilon < 1/(p.t)$. Set $X_i := a + \varepsilon$ for all $i \in \mathcal{I}$. Set $Y_j := a/2 + (j-1)(\varepsilon/t)$ for all $j \in \{1, \ldots, t+1\}$. Set $Y_j := a/2 + (t - (j - (t+2)))(\varepsilon/t) = a/2 + (2t - j2)(\varepsilon/t)$ for all $j \in \{t+2, \ldots, 2(t+1)\}$. Set $F = (a+\varepsilon)(t+1)$. Let $G = (V, E)$ be any graph and let $G' = (V', E')$ be such that $E' = \{\{v'_j, v'_{j+1}\} | j = 1, \ldots, |\mathcal{J}| - 1\} \cup \{\{v'_{|\mathcal{J}|}, v'_1\}\}$. Observe that G' is a cycle. Let $p \geq t$ be any integer. We now define the unitary cost as follows. First, we denote by $h(j)$ the index of the unique node $v'_{h(j)}$ that is at distance $t+1$ of node v'_j in G', for all j, $1 \leq j \leq |\mathcal{J}|/2$. Set $c_{i,i} = c_{i,h(i)} = 1$ for all $i \in \mathcal{I}$ and set $c_{i,j} = p$ for all $i \in \mathcal{I}$ and for all $j \in \mathcal{J} \setminus \{i, h(i)\}$. Set $C = (t+1)(a + \varepsilon p)$.

We now prove that there exists an admissible solution if and only if $M \geq |\mathcal{I}|(|\mathcal{I}| + 1) = (t+1)(t+2) = (|\mathcal{J}|/2)(|\mathcal{J}|/2 + 1)$.

($\Leftarrow$) Assume that $M = (t+1)(t+2)$. We prove that there exists an admissible solution. Consider the following flow. For all $i \in \mathcal{I}$, $f_{i,i} := a/2$, $f_{i,j} := \varepsilon/t$ for every $j \in \{i+1, \ldots, t+1\}$, $f_{i,j} := \varepsilon/t$ for every $j \in \{t+2, \ldots, h(i)-1\}$, $f_{i,h(i)} := a/2$, and $f_{i,j} := 0$ for every $j \in \{h(i)+1, \ldots, |\mathcal{J}|\}$. Every node $u \in V$ sends flow to a sub-path of size $t+2$ of G'. Thus, the connectivity constraints are satisfied and the number of edges that support flow is exactly $(t+2)(t+1) = M$ because $|\mathcal{I}| = (t+1)$. Furthermore, by construction, each node $v'_j \in V'$, $j \in \mathcal{J}$, receives a flow of Y_j. Thus, $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} = F = \sum_{j\in\mathcal{J}} Y_j$. Finally, $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} c_{i,j} = \sum_{i\in\mathcal{I}} a + \varepsilon p = (t+1)(a + \varepsilon p)$ by construction of f.

($\Rightarrow$) Assume that there exists an admissible solution. We prove that $M \geq (t+1)(t+2)$. By contradiction. Suppose that $M < (t+1)(t+2)$. It means that there exists $i^* \in \mathcal{I}$ such that $\sum_{j\in\mathcal{J}|f_{i^*,j}>0} 1 < t+2$ because $|\mathcal{I}| = t+1$. For such a node $v_{i^*} \in V$, it is so impossible to send flow to $v'_{i^*} \in V'$ and flow to $v'_{h(i^*)} \in V'$. Indeed, since any shortest path between v'_{i^*} and $v'_{h(i^*)}$ is composed of t nodes, if node v_{i^*} sends flow to both v'_{i^*} and $v'_{h(i^*)}$, then v_{i^*} must send flow to at least t other nodes because otherwise the connectivity constraints would not be satisfied. Thus, since $\sum_{j\in\mathcal{J}|f_{i^*,j}>0} 1 < t+2$, then either v'_{i^*} or $v'_{h(i^*)}$ does not receive flow from v_{i^*}. Without loss of generality, suppose that such a node is v'_{i^*}. By construction of the instance, it means that $\sum_{i\in\mathcal{I}\setminus\{i^*\}} f_{i,i^*} c_{i,i^*} \geq \frac{p.a}{2}$. We get that $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} f_{i,j} c_{i,j} \geq \frac{p.a}{2} + t.a$ because $\sum_{j\in\mathcal{J}\setminus\{i^*\}} f_{i,j} c_{i,j} \geq t.a$ (the other nodes of G' receive at least a volume of flow of $a/2$ and the cost unitary cost is at least 1). Since the solution is admissible, then we get that $\frac{p.a}{2} + t.a \leq C = (t+1)(a + \varepsilon p)$. Set $p := t$, $a := t$, and $\varepsilon := \frac{1}{p.t}$. We get that $t^2/2 \leq t + 1 + 1/t$. It is clearly false for sufficient large value of t. Thus, the solution is not admissible and we get a contradiction. Thus, it means that $M \geq (t+1)(t+2)$. □

Proof of Lemma 4. Consider the instance I_1 of the proof of Lemma 3 and the instance I_2 of the proof of Lemma 2. We construct the instance I that is the union of I_1 and I_2 such that the unitary cost between any supply (demand, respectively) node of I_1 and any demand (supply, respectively) node of I_2 is infinite or sufficiently large. We set $M = M_{I_1} + M_{I_2} = (t+1)(t+2) + 3m$. We also set $F = F_{I_1} + F_{I_2}$ and $C = C_{I_1} + C_{I_2}$. (The index I_x means that the parameter concerns the instance $x \in \{1, 2\}$.) By Lemma 3 and Lemma 2 (and Corollary 1), we get that the problem

of deciding if there exists an admissible solution is NP-complete. Finally, the number of nodes of I is $|V \cup V'| = 3(t+1) + 4m$. By setting $t = m-1$, we get that $M \in \Theta(|V \cup V'|^2)$. □

Proof of lemma 5. Suppose there is a constant $k > 1$ such that there is a polynomial time k-approximation algorithm for `EMD-FCC`.

Consider an instance of 3-Partition problem. Let $m \geq 1$ be any integer and let $S = \{n_1, n_2, \ldots, n_{3m}\}$ be a set of $3m$ positive integers. Set $|\mathcal{I}| = 3m+1$ and $\mathcal{I}^- = \mathcal{I} \setminus \{3m+1\}$. Set $|\mathcal{J}| = m+1$ and $\mathcal{J}^- = \mathcal{J} \setminus \{m+1\}$. Set $X_i = n_i$ for all $i \in \mathcal{I}^-$. Let $Z = \sum_{i \in \mathcal{I}^-} X_i$. Without loss of generality, let $Y_j = Y$ with $Z = mY$ for all $j \in \mathcal{J}^-$. Set $X_{3m+1} = Z$ and $Y_{m+1} = Y$. Set $c_{i,j} = 1$ for all $i \in \mathcal{I}^-, j \in \mathcal{J}^-$. Set $c_{3m+1,m+1} = 1$. Set $c_{3m+1,j} = K = k(Y+Z)$ for all $j \in \mathcal{J}^-$. Set $c_{i,m+1} = K = k(Y+Z)$ for all $i \in \mathcal{I}^-$. Let $G = (V, E)$ be any connected graph and let $G' = (V', V' \times V')$. The connectivity constraints are always satisfied because G' is a complete graph. Let $F = Z$, $M = 3m+1$, and $D = m+1$.

There exists a solution for `EMD-FCC` such that $\sum_{j \in \mathcal{J}^-} f_{3m+1,j} + \sum_{i \in \mathcal{I}^-} f_{i,m+1} = 0$ if and only if there is a solution for the instance of 3-Partition problem (Lemma 2). The cost of this solution is $Y + \sum_{i \in \mathcal{I}, j \in \mathcal{J}} f_{i,j} c_{i,j} = Y + Z$. We prove that if there does not exist a solution for the instance of 3-Partition problem, then the cost of any admissible solution for `EMD-FCC` is at least $Z - 1 + K$. Suppose that there does not exist a solution for the instance of 3-Partition problem. Thus, we have $\sum_{j \in \mathcal{J}} f_{3m+1,j} > 0$. There are two cases.

- If $f_{3m+1,m+1} = 0$, then $\sum_{i \in \mathcal{I}^-} f_{i,3m+1} = Y$. Thus, we get $\sum_{i \in \mathcal{I}^-} \sum_{j \in \mathcal{J}^-} f_{i,j} \leq Z - Y$ and $\sum_{j \in \mathcal{J}^-} f_{3m+1,j} \geq Y$. Then, $\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} \geq 2YK + Z - Y \geq Z - 1 + K$.
- If $f_{3m+1,m+1} > 0$, then $\sum_{i \in \mathcal{I}^-} \sum_{j \in \mathcal{J}^-} f_{i,j} \leq Z - 2$. Thus, we get $\sum_{j \in \mathcal{J}^-} f_{3m+1,j} \geq 2$. Then, $\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{i,j} c_{i,j} \geq 2K + Y + Z - 2 \geq Z - 1 + K$.

Since $K = k(Y+Z)$, we have $\frac{Z-1+K}{Z} > k$. As we have supposed that there exists a polynomial time k-approximation algorithm for `EMD-FCC`, then if there is a solution of cost $Y + Z$, the k-approximation algorithm returns such a solution (otherwise the approximation ratio would be wrong); otherwise (solution of cost at least $Z - 1 + K$), the k-approximation ratio would return a solution with cost at least $Z - 1 + K$. Thus, the polynomial time (k-approximation) algorithm solves 3-Partition problem which is a strongly NP-complete problem [7]. A contradiction, unless P=NP. □

7.3 Proofs for algorithms – section 4

Proof of Lemma 6. First case. The connectivity constraints are always satisfied because G' is a complete graph. Indeed, for every $H = \{H_1, H_2\}$ such that $|H_1| = |H_2| = 1$, $H_1 \neq H_2$, then $d_{G'}(H_1', H_2') = d_{G'}(cc(H_1')) = d_{G'}(cc(H_1')) = 0$, where H_i' is the sub-graph induced by the nodes that receive flow from the unique node that belongs to H_i, $i \in \{1, 2\}$. Furthermore, the transport size constraint is also always satisfied because $M = |E(B)|$. Thus, `EMD-FCC` and `EMD-CCC` can be solved by linear programs that are variants of the linear program described in Eq. 1 for `EMD`.

Second case. The connectivity constraints are always satisfied because G' has diameter at most $g(x)$ for all $x \geq 0$. Indeed for every $H = \{H_1, H_2\}$ such that $|H_1| = |H_2| = 1$, $H_1 \neq H_2$, then $d_{G'}(H_1', H_2') \leq \max_{u', v' \in V'} d_{G'}(u', v') \leq g(d_G(H_1, H_2))$ and $d_{G'}(cc(H_i')) \leq \max_{u', v' \in V'} d_{G'}(u', v') \leq g(0)$, for any $i \in \{1, 2\}$. Furthermore, the transport size constraint is also always satisfied because $M = |E(B)|$. Thus, `EMD-FCC` and `EMD-CCC` can be solved by linear programs that are variants of the linear program described in Eq. 1 for `EMD`. □

Proof of lemma 7. In this proof, F is always chosen as the sum of the volumes of demands. Consider an instance of `EMD-FCC`. We construct an auxiliary instance as follows. The graphs G,

G', and B and the cost $c_{i,j}$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$ are those of the original instance. Let $\varepsilon' > 0$ be a real value such that $X_i - |\mathcal{J}|\varepsilon' > 0$ for all $i \in \mathcal{I}$ and such that $Y_j - |\mathcal{I}|\varepsilon' > 0$ for all $j \in \mathcal{I}$. We denote by X'_i the volume of supply for all $i \in \mathcal{I}$ in the auxiliary instance. Set $X'_i = X_i - |\mathcal{J}|\varepsilon'$ for all $i \in \mathcal{I}$. We denote by Y'_i the volume of demand for all $j \in \mathcal{J}$ in the auxiliary instance. Set $Y'_j = Y_j - |\mathcal{I}|\varepsilon'$ for all $j \in \mathcal{I}$. Let f' be an optimal solution for this auxiliary instance for `EMD`. Recall that this can be done in polynomial time since it reduces to solve a linear program. The cost of f' is $\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j}$.

We now construct an admissible solution f for the original instance for `EMD-FCC` as follows. Set $f_{i,j} = f'_{i,j} + \varepsilon'$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$. All the connectivity constraints are satisfied because for every $v \in V$, we have $H'_v = G'$, where H'_v is the sub-graph induced by the of nodes of G' that receive flow from v. Recall that $M = |E(B)|$. Thus, the solution is admissible. The cost of f is $\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f_{i,j}c_{i,j} = |\mathcal{I}||\mathcal{J}|\varepsilon' + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j}$. Let f^* be an optimal solution for the original instance of `EMD-FCC`. Observe that

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j} \leq \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f^*_{i,j}c_{i,j} \text{ and that}$$

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f^*_{i,j}c_{i,j} \leq |\mathcal{I}||\mathcal{J}|\varepsilon' + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j} = \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f_{i,j}c_{i,j}.$$

We finally choose $\varepsilon' > 0$ such that

$$|\mathcal{I}||\mathcal{J}|\varepsilon' + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j} \leq (1+\varepsilon)\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f^{'}_{i,j}c_{i,j}.$$

Thus, we get

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f_{i,j}c_{i,j} = |\mathcal{I}||\mathcal{J}|\varepsilon' + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j} \text{ and}$$

$$|\mathcal{I}||\mathcal{J}|\varepsilon' + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f'_{i,j}c_{i,j} \leq (1+\varepsilon)\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} f^*_{i,j}c_{i,j}.$$

We get a polynomial time $(1+\varepsilon)$-approximation algorithm for `EMD-FCC` because f' is obtained by solving a linear program and f is directly deduced from f'. □

Proof of lemma 8. The number of steps (number of iterations of the while loop of Algorithm 1) is at most $|V||V'|$. At the beginning of each step, we select an edge. The time-complexity of Line 5 of Algorithm 1 to perform such edge selection is $O(|V||V'|)$. We get the first term $O(|V|^2|V'|^2)$. At the end of each step, we update $b_{i,j,}$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$. The time complexity of such a computation is $C(g)$. We get the second term $O(|V||V'|C(g))$. To conclude, note that other computations are negligible. □

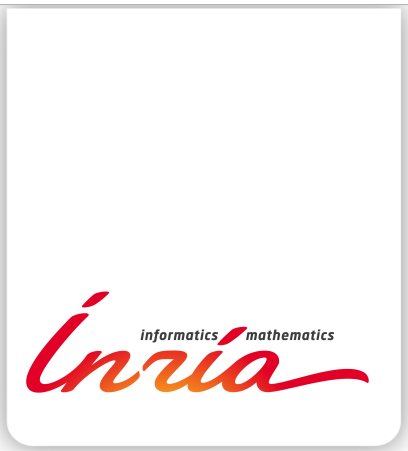
Proof of lemma 10. Before calling the main algorithm, we can initially compute the shortest paths between any two nodes of G' by using the classical Floyd Warshall algorithm. We also do such a computation for any two nodes of G. The time complexity of such first step is $O(|V|^3 + |V'|^3)$. With such an initial computation, the time complexity of Lines 11-14 of Algorithm 2 is $O(|V'|^2\log(|V'|) + \log(|V|))$. We get that the time complexity of Lines 8-14 is $O(|V|^2|V'|(|V'|^2\log(|V'|) + \log(|V|)))$ and that computation dominates the complexity of Lines 4-7. Then, the global time complexity is $O(\max(|V|^3, |V'|^3, |V|^2|V'|(|V'|^2\log(|V'|) + \log(|V|))))$ because of the three for loops (Lines 8-10). After simplification, we get that the time complexity is $O(\max(|V|^3, |V|^2|V'|^3\log(|V'|)))$. □

Proof of lemma 11. Lines 1-6 of Algorithm 4 update the boolean function b if $X_i - (x_{i_t} + f_{i_t,j_t}) = 0$, that is if the supply node v_{i_t} sends flow for the first time. In that case, all the neighbors of v'_{j_t} in G' can receive flow from v_{i_t} (Line 2) and all the other nodes of G' cannot receive flow from v_{i_t} (Line 3). Furthermore, all the neighbors of v_{i_t} in G that do not have sent flow, cannot send flow to the nodes of G' that are not neighbors of v'_{j_t} in G' (Lines 4-6).

Lines 7-15 update the boolean function b if $X_i - (x_{i_t} + f_{i_t,j_t}) \neq 0$, that is if the supply node v_{i_t} has already sent flow before the current step. All the neighbors of v'_{j_t} in G' can receive flow from v_{i_t} (Line 8). Every neighbor v_i of v_{i_t} in G that does not have sent flow, can send flow to every neighbor v'_j of v'_{j_t} in G' if $b_{i,j} = 0$, that is if the edge $\{v_i, v'_j\}$ does not support flow (Lines 9-13). Furthermore, for every neighbor v_i of v_{i_t} in G that does not have sent flow, for every neighbor v'_j of v'_{j_t} in G' such that $b_{i,j} = 0$, and for every neighbor v_k of v_i in G that has already sent flow and such that v_k cannot send flow to v'_j, then we set that v_i cannot send flow to v'_j, that is $b_{i,j} = 0$ (Lines 9-15).

Algorithm 4 finally returns the variables $b_{i,j}$ for all $i \in \mathcal{I}, j \in \mathcal{J}$ (Line 16). □

Proof of lemma 12. The time complexity of the first part (Lines 1-6) is $O(|V'| + \Delta(G)\Delta(G'))$, and the time complexity of the second part (Lines 7-15) is $O(\Delta(G)^2\Delta(G'))$. □



RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399